



TITLE:

# 自己組織化マップにおける異型学習と吸収的特性について (非線形解析学と凸解析学の研究)

AUTHOR(S):

星野, 満博

---

CITATION:

星野, 満博. 自己組織化マップにおける異型学習と吸収的特性について (非線形解析学と凸解析学の研究). 数理解析研究所講究録 2013, 1841: 93-98

ISSUE DATE:

2013-07

URL:

<http://hdl.handle.net/2433/194977>

RIGHT:

Absorbing property in self-organizing maps  
with variants of learning process  
(自己組織化マップにおける異型学習と吸収的特性について)

秋田県立大学 システム科学技術学部 星野満博 (Mitsuhiro Hoshino)  
Faculty of Systems Science and Technology, Akita Prefectural University

1. 基本的な自己組織化マップモデル

本報告は Kohonen 型アルゴリズム [7] として知られている自己組織化マップモデルにおける整列化とモデル関数の性質に関する一つの理論的考察である。自己組織化マップモデルにおけるノードの配列とノードの値との間に現れるある種の規則性について、モデルの順序化、整列化の形成過程と状態クラスの閉性に注目する。本報告では、基本的な学習ルールと異なる、ある異型学習ルールを用いた場合において、整列化についての考察および数値例を与える。

自己組織化マップは非常に実用的であり広範囲に応用例を有し、アルゴリズムも非常にシンプルであるが、その数学的構造はあまり明らかではない。

本報告では、自己組織化マップモデルをノード、ノードの値、インプット、学習プロセスの 4 つの要素によって、以下の様に定義する。

$$(I, V, X, \{m_k(\cdot)\}_{k=0}^{\infty})$$

- (i)  $I$  をすべてのノードの集合とする。  $I$  は、距離  $d$  をもつある距離空間の加算部分集合とする。
- (ii) 各ノードは、それぞれ 1 つの値をもつ。  $V$  をノードの値の空間とする。  $V$  はノルム空間であると仮定する。  $V$  におけるノルムを  $\|\cdot\|$  とする。  $m(i)$  をノード  $i$  の値として、その対応  $m: I \rightarrow V$  をモデル関数と呼ぶことにする。 また、  $M$  をモデル関数の全体、  $m_0: I \rightarrow V$  を初期モデル関数とする。
- (iii)  $X \subset V$  を入力集合とする。  $x_0, x_1, x_2, \dots \in X$  を入力列とする。
- (iv) 学習プロセス

$$m_{k+1}(i) = (1 - \alpha_{m_k, x_k})m_k(i) + \alpha_{m_k, x_k}x_k \quad (1)$$

ここで、  $\alpha_{m_k, x_k}$  は、  $0 \leq \alpha_{m_k, x_k} \leq 1$  を満たす学習率を表す。

例えば、 $n$  個のノード  $1, 2, \dots, n$  がある場合を考える。そのそれぞれに対してノードの値  $m_0(1), m_0(2), \dots, m_0(n)$  が与えられているとする。このとき、入力とこれに伴う学習により各ノードの値が更新される。 $x_0 \in X$  が入力されたならば、

$$m_1(i) = (1 - \alpha_{m_0, x_0})m_0(i) + \alpha_{m_0, x_0}x_0$$

により、ノードの値を更新する。インプット  $x_1, x_2, x_3, \dots$  に対して、これを繰り返すことにより、モデル関数  $m_1, m_2, m_3, \dots$  が逐次に生成され、ノードの更新が更新される。

このような学習を十分な回数、繰り返したとき、モデル関数において、単調性等、各ノードの値の配列にある種の規則性が現れることがある。様々なノード集合、ノードの値の空間、学習方法において、単調化等の幾つかの興味深い現象が現れる。

## 2. $\mathbb{R}$ 値ノード、1次元ノード配列モデルと異型学習

ここでは、最も単純な自己組織化マップモデルである、 $\mathbb{R}$  値ノード、1次元ノード配列の場合について述べる。

- (i) 有限個のノードを仮定する。 $I = \{1, 2, \dots, N\} \subset \mathbb{N}$ .
- (ii) ノード値の空間を  $\mathbb{R}$  (ユークリッドノルム) とする。 $m_0 = [m_0(1), m_0(2), \dots, m_0(n)]$  などと記すことにする。
- (iii)  $x_0, x_1, x_2, \dots \in X \subset \mathbb{R}$  を入力列とする。
- (iv) 学習プロセス  $L_E$  (1次元配列,  $\mathbb{R}$ -値ノード,  $\varepsilon = 1$ )

(a) 学習範囲:

$$I(m_k, x_k) = \left\{ i^* \in I \mid |m_k(i^*) - x_k| = \inf_{i \in I} |m_k(i) - x_k| \right\} \\ (m_k \in M, x_k \in X), \\ N_1(i) = \{j \in I \mid |j - i| \leq 1\} \quad (i \in I).$$

(b) 学習率:  $0 \leq \alpha \leq 1$ .

(c) 更新後の値:

$$m_{k+1}(i) = \begin{cases} (1 - \alpha)m_k(i) + \alpha x_k & \text{if } i \in \bigcup_{i^* \in I(m_k, x_k)} (N_1(i^*) \setminus \{i^*\}), \\ m_k(i) & \text{if } i \notin \bigcup_{i^* \in I(m_k, x_k)} (N_1(i^*) \setminus \{i^*\}), \end{cases} \quad k = 0, 1, 2, \dots$$

学習範囲としてノード  $i^*$  の近傍から  $i^*$  を除いたものを用いる。通常、学習プロセスとしては、以下に述べる  $i^*$  も学習範囲に含めたものを採用する。

基本的な学習プロセスとして次の2つのタイプが挙げられる。それぞれに対して以下の基本的な性質をもつ。

**Theorem 1** 学習プロセス  $L_A$  (1次元配列,  $\mathbb{R}$ -値ノード,  $\varepsilon = 1$ )

(a) 学習範囲:

$$I(m_k, x_k) = \left\{ i^* \in I \mid |m_k(i^*) - x_k| = \inf_{i \in I} |m_k(i) - x_k| \right\}$$

$$(m_k \in M, x_k \in X),$$

$$N_1(i) = \{j \in I \mid |j - i| \leq 1\} \quad (i \in I).$$

(b) 学習率:  $0 \leq \alpha \leq 1$ .

(c) 更新後の値:

$$m_{k+1}(i) = \begin{cases} (1 - \alpha)m_k(i) + \alpha x_k & \text{if } i \in \bigcup_{i^* \in I(m_k, x_k)} N_1(i^*), \\ m_k(i) & \text{if } i \notin \bigcup_{i^* \in I(m_k, x_k)} N_1(i^*), \end{cases} \quad k = 0, 1, 2, \dots$$

を仮定するとき, モデル関数に対して以下が成り立つ.

- (i) モデル関数  $m_k$  が  $I$  上で単調増加であるならば, モデル関数  $m_{k+1}$  も  $I$  上で単調増加である.
- (ii) モデル関数  $m_k$  が  $I$  上で単調減少であるならば, モデル関数  $m_{k+1}$  も  $I$  上で単調減少である.
- (iii) モデル関数  $m_k$  が  $I$  上で狭義単調増加であるならば, モデル関数  $m_{k+1}$  も  $I$  上で狭義単調増加である.
- (iv) モデル関数  $m_k$  が  $I$  上で狭義単調減少であるならば, モデル関数  $m_{k+1}$  も  $I$  上で狭義単調減少である.

**Theorem 2** 学習プロセス  $L_m$  (1次元配列,  $\mathbb{R}$ -値ノード,  $\varepsilon = 1$ )

(a) 学習範囲:

$$J(m_k, x_k) = \min \left\{ i^* \in I \mid |m_k(i^*) - x_k| = \inf_{i \in I} |m_k(i) - x_k| \right\}$$

$$(m_k \in M, x_k \in X),$$

$$N_1(i) = \{j \in I \mid |j - i| \leq 1\} \quad (i \in I).$$

ここで, 上の  $\min$  は, 自然数の順序の意味での最小値を意味する.

(b) 学習率:  $0 \leq \alpha \leq 1$ .

(c) 更新後の値:

$$m_{k+1}(i) = \begin{cases} (1 - \alpha)m_k(i) + \alpha x_k & \text{if } i \in N_1(J(m_k, x_k)), \\ m_k(i) & \text{if } i \notin N_1(J(m_k, x_k)), \end{cases} \quad k = 0, 1, 2, \dots$$

を仮定するこき、モデル関数に関して、次が成り立つ。

- (i) モデル関数  $m_k$  が  $I$  上で狭義単調増加であるならば、モデル関数  $m_{k+1}$  も  $I$  上で狭義単調増加である。
- (ii) モデル関数  $m_k$  が  $I$  上で狭義単調減少であるならば、モデル関数  $m_{k+1}$  も  $I$  上で狭義単調減少である。

ここでの単調増加性、単調減少性のように、モデル関数が一度その状態になると、その状態が保存されるという意味において、このような状態を自己組織化マップモデルの吸収状態と呼ぶことにする。

### 3. 異型学習における整列化について

学習のタイプとして、 $L_E(\varepsilon = 1)$  を用いた場合について考える。  $L_E$  を用いたとき、 $L_A$  や  $L_m$  を用いた場合と異なり、単調性は保存されるとは限らない。しかし、以下のように  $\alpha$  を十分小さくすることにより単調性を保存することができる。

**Theorem 3** 学習プロセス  $L_E(\varepsilon = 1)$  において、以下の学習率  $\alpha$  を用いることを仮定する。

$$I(m_k, x_k) = \left\{ i^* \in I \mid |m_k(i^*) - x_k| = \inf_{i \in I} |m_k(i) - x_k| \right\},$$

$$i_m = \min I(m_k, x_k),$$

$$i_M = \max I(m_k, x_k),$$

$$0 \leq \alpha \leq \alpha_1$$

ただし、与えられた  $\alpha_0$  ( $0 < \alpha_0 \leq 1$ ) に対して、

$$i_m < i_M \text{ のとき } \alpha_1 = \alpha_0,$$

$$i_m = i_M \text{ のとき}$$

$$\alpha_1 = \min \left\{ \alpha_0, \frac{\min\{|m_k(i_m) - m_k(i_m - 1)|, |m_k(i_m + 1) - m_k(i_m)|\}}{|m_k(i_m + 1) - m_k(i_m - 1)|} \right\}$$

とする。ここで、 $i_m = i_M = 1$  のとき、 $m_k(i_m - 1) = x$ ,  $i_m = i_M = N$  のとき、 $m_k(i_m + 1) = x$  とする。また、上の分数式において、分母が 0 となる場合は、 $\min$  として  $\alpha_0$  を採用することとする。また、上の  $\min I(m_k, x_k)$ ,  $\max I(m_k, x_k)$  は、自然数の順序の意味での最小値、最大値を意味する。

このとき、モデル関数に関して、次が成り立つ。

- (i) モデル関数  $m_k$  が  $I$  上で単調増加であるならば、モデル関数  $m_{k+1}$  も  $I$  上で単調増加である。
- (ii) モデル関数  $m_k$  が  $I$  上で単調減少であるならば、モデル関数  $m_{k+1}$  も  $I$  上で単調減少である。

このとき、学習率が小さいので整列化するまでに時間が掛かることに注意する。応用上は、このような場合、未修整の学習率を用いて十分な回数の学習をさせ、十分整列化した後に上記の様に学習率を小さくすればよい。

#### 4. EXAMPLE

##### Example 1

次のような、35個の1次元配列された実数値ノードからなる自己組織化マップの数値計算の例を与える。ノード集合を

$$I = \{1, 2, 3, \dots, 35\}$$

とする。ノードの初期値はランダムなもの（0以上、10以下の整数）を用い、また、入力列は  $[0, 10]$  上の一様分布に従う確率変数によって生成させたものを使用している。学習プロセスは、 $L_E(\varepsilon = 1)$  および定理3における学習率  $\alpha = \alpha_1$  を用いる。ここで、学習率の上限は  $\alpha_0 = 0.5$  を用いている。ノードの初期値及びこれらを学習により300回、1000回、15000回更新させたときの値は、図1のようになった。横軸はノードを表し、縦軸はノードの値を表す。例えば、ノード1及び2の初期値は、それぞれ9、0である。数百回程度の更新により、徐々に凹凸が減り、約12000回の更新によりモデル関数が単調化され、以後、単調性が保存されている。入力値を離散の値にした場合、特に、入力値の取りうる値が少ない場合において、一定の回数の学習の後、モデル関数の変化が著しく減少する傾向にある。

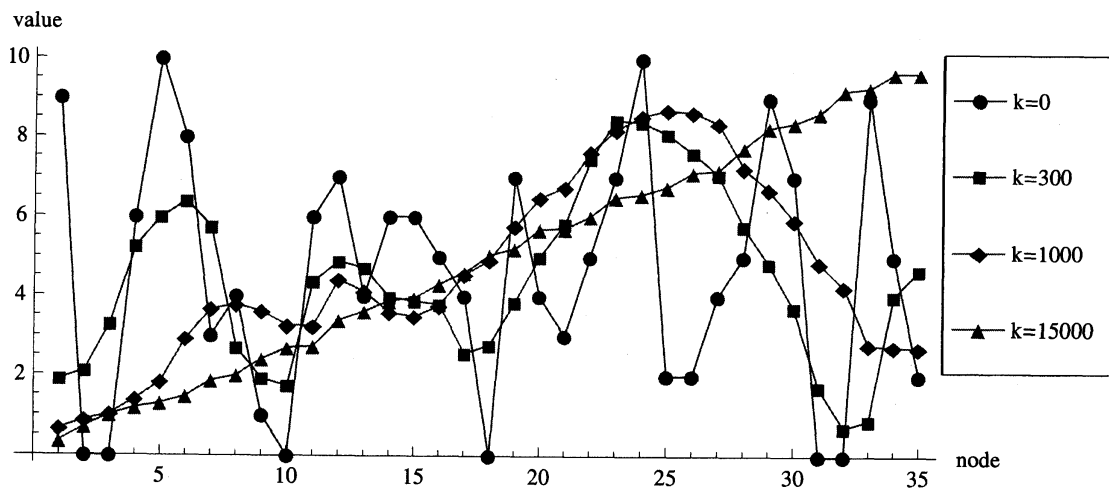


図1: ノードの初期値と更新後の値 ( $k$ は更新回数: 300回, 1000回, 15000回)

□

## 参考文献

- [1] M. Cottrell and J.-C. Fort, *Étude d'un processus d'auto-organisation*, Annales de l'Institut Henri Poincaré, 23(1) (1987), pp.1–20 (in French)
- [2] E. Erwin, K. Obermayer, and K. Schulten, *Convergence properties of self-organizing maps*, In T. Kohonen, K. Mäkisara, O. Simula, and J. Kangas, editors, Artificial Neural Networks, Amsterdam Netherlands Elsevier (1991), pp.409–414
- [3] E. Erwin, K. Obermayer and K. Schulten, *Self-organization maps: stationary states, metastability and convergence rate*, Bio. Cybern., 67 (1992), pp. 35–45.
- [4] E. Erwin, K. Obermayer and K. Schulten, *Self-organization maps: ordering, convergence properties and energy functions*, Bio. Cybern., 67 (1992), pp. 47–55.
- [5] W. Fujiwara, E. Itou, M. Hoshino, I. Kaku, A. Sakusabe, M. Sasaki and H. Kosaka, *A study on the effective method of external inspecting using a neural network approach*, Proceedings of 6th ICIM (2002), pp. 369–375
- [6] M. Hoshino and Y. Kimura, *Ordered states and probabilistic behavior of self-organizing maps*, Nonlinear Analysis and Optimization (Shimane, 2008), Yokohama Publishers, pp. 31–44
- [7] T. Kohonen, *Self-Organizing Maps, Third Edition*, Springer, 2001.
- [8] P. L. Zador, *Asymptotic quantization error of continuous signals and the quantization dimension*, IEEE Trans. Inform. Theory, vol. IT-28, No. 2, March (1982), pp. 139–149.